講座
Seminar

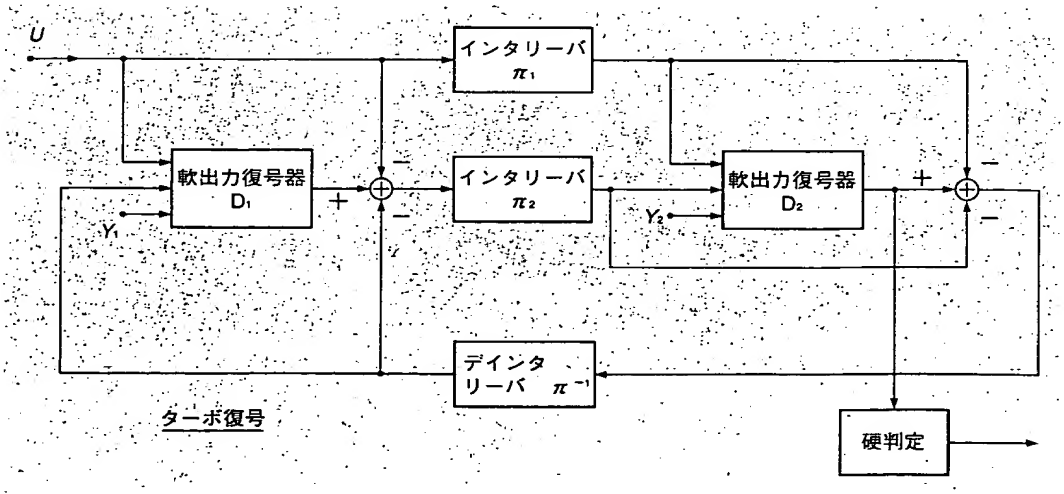
シャノン限界に迫る新しい符号化方式「ターボ符号」

移動体通信への応用が確実に

山口 和彦
電気通信大学 情報工学科 講師

今井 秀樹
東京大学 生産技術研究所 教授

符号理論の世界が新しい話題で活気づいている。シャノンの通信路符号化定理によって与えられる，誤りなしに送信可能な伝送速度の理論上の限界，いわゆる「シャノン限界」に一歩近づいた新しい符号が登場したからである。生まれ故郷はフランス。研究発表の数は年を追って増えている。移動体通信への応用は確実になった。通信分野に加え，磁気記録の再生系への適用なども検討が始まった。（本誌）



ターボ（turbo）符号，ターボ復号と呼ばれる誤り訂正方式がシャノン（Shannon）限界に迫る新しい符号化・復号方式として近年，符号理論や通信分野の研究者のあいだで注目を浴びている。シャノン限界とは，誤りなしに送信可能な伝送速度の理論上の限界である（詳細は後述）。ターボ符号の研究は，フランスのC.Berrouらが ICC '93（通信分野の国際会議，1993 IEEE International Conference on Communications）で行なった発表で始まった[1]。

応用研究の発表件数は増加している。そのなかでも，移動体通信への応用は，ほぼ確実になった。ターボ符号がフェージング（電波の強度がめまぐるしく変動する状態）に強いからである。ターボ符号の使用で高い符号化利得が望める放送分野も，期待できる応用先である。

符号器・復号器（コーデック）は，ターボ符号を生んだフランスで通信機器メーカがすでにLSI化している。

ターボ・エンジンに見立てる

方式を簡単に説明しよう（図1）。送信側はデータを通常二つの畳込み符号により符号化する（各符号の意味については，p.167の「符号の種類を整理する」を参照）。ブロック符号では一つのブロック（符号語）ごとに符号化処理が完結する。これに対し，畳込み符号では後続するブロックにも影響を及ぼすのが特徴であり，衛星通信などでよく使われている。ターボ符号化においては2番目の畳込み符号の符号化の前に一度メモリにデータを書き込み，これを異なる順で取り出すインタリーバによりデータの順序を攪拌（かくはん）する。受信側ではこの二つの畳込み符号に対する復号器により送信データを推定する。このとき，一方の復号器の結果をフィードバックしながら繰り返し復号を行なう。ここが特徴である。

ターボ・エンジンは，燃焼により推進力を得るだけでなく，その排気ガス流の圧力を用いてターボ・チャージャのタービンを回す。その力により空気と燃料の混合気の圧力を高め，燃焼効率を向上させる。こうすることで，エンジン性能を高める。ターボ復号は，送信された符号語と誤りの混合気（受信語）を燃焼（復号）させるのに排気ガス（復号結果）を入力側にフィードバックして復号性能を上げる。この操作を，ターボ・エンジンと見立てたものだ。C.E. Shannonが情報理論の研究を進めたとき，情報量の定義に熱（統計）力学のエントロピの概念を比喩的に取り入れた。そのことも思い起

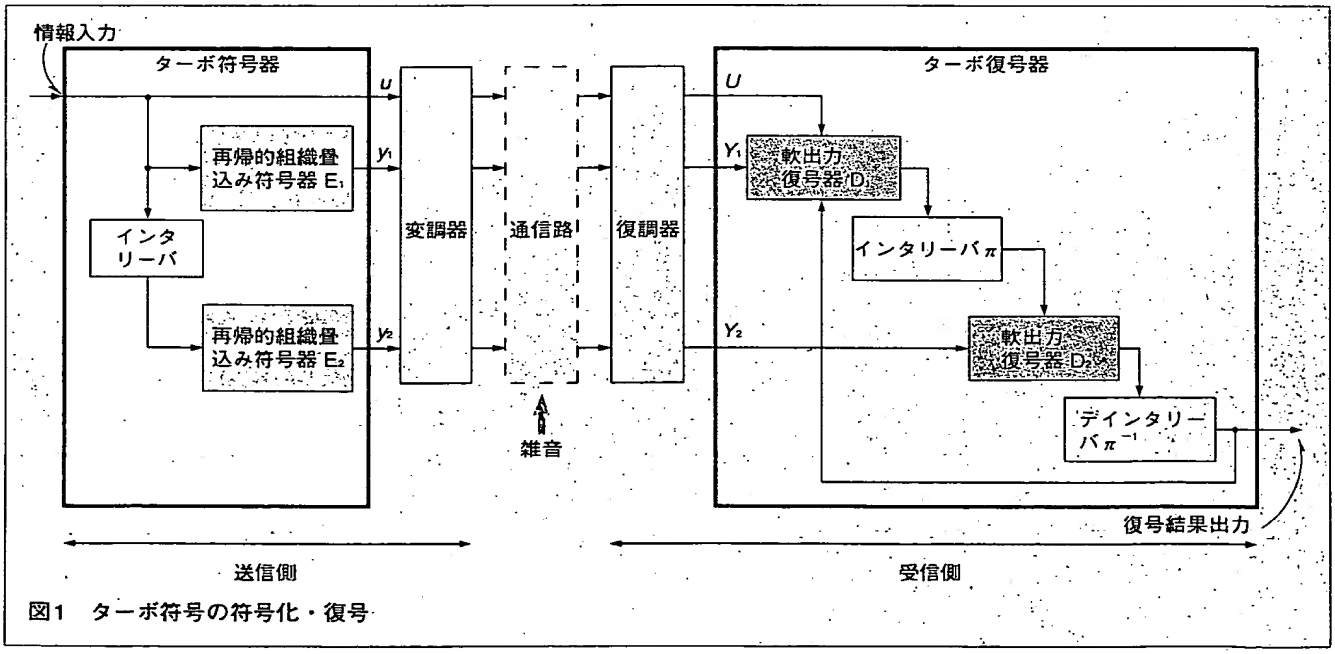


図1　ターボ符号の符号化・復号

こさせる，センスのよい命名といえる。

初めは半信半疑

ターボ符号が発表された初めのころは，示された特性に疑いをもつ研究者も多かった。これは，Berrouらの提案では，方式と性能の紹介はあったものの，理論的な考察がほとんど行なわれていなかったこと，また既存の方法の組み合わせに過ぎず新規性を欠くと思われたことなどによる。

事実，ターボ符号化・復号の主な要素として用いられる軟出力復号（p.170の「ターボ復号に至る，符号化と復号の歴史」参照），インタリーバや再帰的組織畳込み符号（**表1**）は，決して目新しいものではない。軟出力復号は，復号結果の信頼度情報を復号結果に添えて出力する方式であるが，連接符号の復号にすでに提案された方式である。また，インタリーバは，移動体通信などで当たり前に用いられている。さらに，再帰的組織畳込み符号も，後述するようにフィードバックのある符号器を採用することで，組織畳込み符号と非組織畳込み符号の両者の長所を活かした符号である。これは，V.32以降のモデムに用いられているトレリス符号に似ている。

しかし，発表後2，3年のうちに興味をもつ研究者が増え，追試や理論的な研究が進んだ。ICCや，GLOBECOM（IEEE Global Telecommunications Conference）といった通信関係者を中心とした会議のみならず，ISIT（情報理論や符号理論を中心とする国際会議，IEEE International Symposium on Information Theory）でも，1995年からターボ符号専門のセッションが設けられている。さらに，1996年スウェーデンでTurbo Coding Seminar[2]が，1997年フランスでInternational Symposium on Turbo Codes and Related Topics[3]というターボ符号に特化したシンポジウムが開かれるなど，ますます盛んになってきた。

そのなかで，ターボ符号化・復号がこれまでにないほどシャノン限界に迫る優れた性能をもつ方式であることが，他の研究グループによっても確認された。ターボ符号，復号方式に関する理論的な評価はさほど進んでいないものの，すでにターボ符号・復号器の実用を目指したLSIも商品化された。さらに通信路の等化にターボ復号を用いるターボ・イコライザ[4),5)]，磁気記録への応用[6)]といった研究も始まっている。

国内では1996年ころよりターボ符号関連の研究会などの企画が進められ，関係する研究発表も増えている。1996年12月に電子情報通信学会情報理論研究会[7),8)]と，情報理論とその応用学会のシンポジウムが連続する日程で開催され，時には，夜遅くまで白熱した討議が行なわれた。翌年の1997年，電子情報通信学会基礎・境界ソサイエティ大会でシンポジウムの企画が組まれた[9)]。また実用面でも，郵政省やNTT移動通信網などが次世代移動体通信への適用の可能性を検討している[10)]。

表1　再帰的組織畳込み符号の特徴

符号	自由距離	特徴
（通常の）組織畳込み符号	小	送信データ自身も伝送。
非組織畳込み符号	大	伝送さるのは送信データ自身ではなく，その符号化系列のみ。
再帰的組織畳込み符号	大	送信データ自身も伝送。終端処理が必要。

再帰的組織畳込み符号は，同じ拘束長で任意の非組織畳込み符号に等しい自由距離†の符号が実現できる。低*S/N*で利用する場合に，非組織畳込み符号に比べビット誤り率特性に優れる。一方，再帰構造のため，符号器の内部状態を初期状態に戻す終端処理が必要であるなど，再帰構造のない符号と応用上異なる注意点がある。

超えられない壁：シャノン限界

シャノン限界は，シャノン（C.E.Shannon）の通信路符号化定理（p.172の「シャノンの通信路符号化定理」参照）により与えられる，誤りなしに送信可能な伝送速度の理論上の限界である。通信路符号化定理は，通信路容量を超えない

†最小距離，自由距離＝ブロック符号の符号語間の最小距離（通常ハミング距離を用いる）は，誤り訂正符号の訂正能力を測る最も簡単な尺度である。最小距離が大きいほど誤りに耐性がある。畳込み符号では慣例上これを最小自由距離，または単に自由距離と言う。通信路の誤り確率が十分小さい場合，復号誤り率特性はほぼ最小距離あるいは自由距離で決まる。

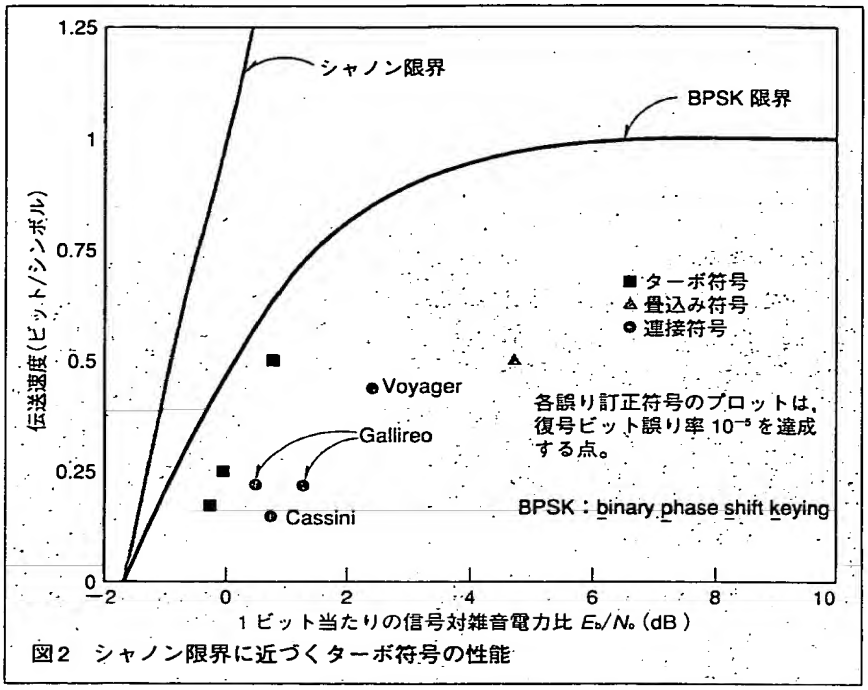


図2　シャノン限界に近づくターボ符号の性能

データ速度であれば，誤りの確率をいくらでも0に近づける符号化方法が存在する，という定理である。通信路容量とは，変復調方式や受信端における信号対雑音電力比（S/N）などにより定まる通信路固有の値のことである。

ここでは，最も基本的なディジタル変調方式である，符号0，1の値を正弦波の位相の違いで表す2相位相変調方式（BPSK：binary phase shift keying）を例としてとりあげる。仮定を二つおく。受信機には，送信された正弦波の周波数を誤差なく知り，その位相を復元する同期検波を仮定する。また，雑音は，周波数によらず振幅の大きな雑音ほど起こりにくい，すなわちその電力が正規分布で発生する白色ガウス雑音（AWGN：additive white Gaussian noise）を仮定する。白色ガウス雑音通信路は最も単純なモデルだが，衛星通信の環境によく一致することがわかっている。S/Nに対応する量として，ここでは情報1ビット当たりに換算した信号対雑音電力比E_b/N_0を用いる。

E_b/N_0を横軸，伝送速度（単位はビット/シンボル：bit/symbol）を縦軸とすると，通信路容量は図2にBPSK限界と記されている曲線により与えられる。ビット/シンボルを単位とする伝送速度は，0，1を表す一つの信号波形（シンボル）当たりの送信情報ビット数である。ただし，ここで言う送信情報とは，送られた正味の情報を意味し，誤り訂正のために付加されたビットは含まない。

この曲線の上にくるようなE_b/N_0と伝送速度の組み合わせに対しては，BPSKを用いる限り，送信情報を正しく送ることが不可能である。この通信路容量の曲線がBPSKの場合のシャノン限界（BPSK限界）である。一つの信号波形が表す記号が3種類以上になる信号を用いる多値変調の場合は，通信路容量は増加する。あらゆる変調を考えた場合の総合的な限界が図2のシャノン限界である。

図2に示す通り伝送速度が0.5ビット/シンボルの場合，BPSK限界は0.2dBである。これに対し，例にあげたターボ符号は0.7dBを達成し，その差は0.5dBに迫っている。衛星通信で用いられる拘束長$K=7$の畳込み符号（p.167の「符号の種類を整理する」参照）では4.6dBにすぎないし，惑星探査機Voyagerで用いた，畳込み符号とリードソロモン（Reed-Solomon）符号（以下RS符号）とを組み合わせた連接符号（0.43ビット/シンボル）でも2.5dBである。

まだ改善の余地がある

ただし，連接符号よりもターボ符号が優れると結論づけることはできないことに注意したい。言えることは，現在のところ，ターボ符号がシャノン限界に最も近い結果を達成しているということだけである。どちらも改善の余地があり，実用上も長所短所があるのだ。

図2に惑星探査機Gallireoの例が二つ出ているが，これは発射後，送信機の不調により符号化，復号方式を変更したためである。どちらもRS符号と畳込み符号の連接符号を用いた方式であるが，左側の特性の良い方のデータは，畳込み符号とパケット・サイズを変更

し，ターボ符号と同じように繰り返し復号を行なって性能を向上させた結果である。

このようなRS符号と畳込み符号の連接符号は現在では，ディジタル放送や衛星通信の主要な符号化方式となりつつあり，普及が進んでいる。

ターボ符号，ターボ復号はシャノン限界に迫る優れた特性を持つ方式であるが，図1を用いて簡単に方式を説明したように，さまざまな技術の組み合わせにより実現されている。

以下では，図1に概略を示したターボ符号化と復号について，原理を説明しよう[1),11),14)]。

ターボ符号化，鍵は二つ

ターボ符号化では，再帰的組織畳込み符号化の採用，非一様インタリーバの採用が鍵となる。それらについて解説する。

複数の要素符号により符号化

ターボ符号化は，インタリーバを介して複数の誤り訂正符号（以下，要素符号と呼ぶ）で符号化することにより実現する。要素符号は，軟出力復号（後出のターボ復号で説明）が可能であればよく，

符号の種類を整理する

符号には，2元符号ー多元符号，ブロック符号ートレリス符号（畳込み符号を含む），組織符号ー非組織符号という大きな分類がある。

2元符号は0，1を符号化の単位とする符号であり，**多元符号**はたとえばバイト（byte）を処理の単位とする。

kシンボル（2元符号ならビットと言い換えてもいい）の情報から，誤り訂正を可能とする構造を付加したnシンボルのブロック（符号語）を得る。これを符号化といい，この符号を，**(n，k) ブロック符号**という。ブロック符号では符号化の処理がこの一つのブロックで完結する。一方，**トレリス符号**は，入力される系列を随時処理していくことで符号化する。2元符号の場合，毎時刻kビット入力し，M時刻前までの$k(M+1)$ビットの情報からnビットを出力すれば，情報1ビットは，$K=n(M+1)$ビットに影響を及ぼす。この符号を，符号化率k/n，拘束長Kのトレリス符号という。これは，すべての符号語が格子（トレリス，trellis）を利用して表現できるためである。**畳込み符号**はトレリス符号の中で限られた演算（線形演算）のみを利用する符号である。後述するビタビ復号や，SOVA（soft-output Viterbi algorithm），MAP復号（maximum a posteriori probability decoding）はこのトレリスを利用して復号されるので，その計算量はトレリスの複雑さに依存する。拘束長が大きくなると，自由距離†が増すが，トレリスが複雑になり，ビタビ復号の計算量は指数関数的に増大する。なお，拘束長の定義は，符号器のメモリ数で定義することもある。これはしばしばνと書くことが多い。

組織符号は符号語nビットのなかに入力されたkビットの情報がそのまま現れる符号である。言い換えれば，組織符号は，符号器に入力された情報ビットに誤り訂正のためのビットを付加する，という形で符号化される符号である。通常，ブロック符号には組織符号を使う。ディジタル・オーディオなどでよく使用する**リードソロモン（RS）符号**は，多元のブロック符号である。

畳込み符号には2元の**非組織符号**がよく使われる（p.165の表1参照）。畳込み符号以外のトレリス符号は，モデムなどで変調方式と一体化して用いることが多い。誤り訂正符号は個々の符号をそのまま利用することもあるが，積符号，連接符号のように符号を組み合わせて用いることも多い。RS符号と畳込み符号の**連接符号**は，はじめに情報をRS符号で符号化し得られたシンボルをビット列に並べ替えて，2元畳込み符号により符号化し送信する。復号は，逆に畳込み符号の復号器，RS符号の復号器で行なう。

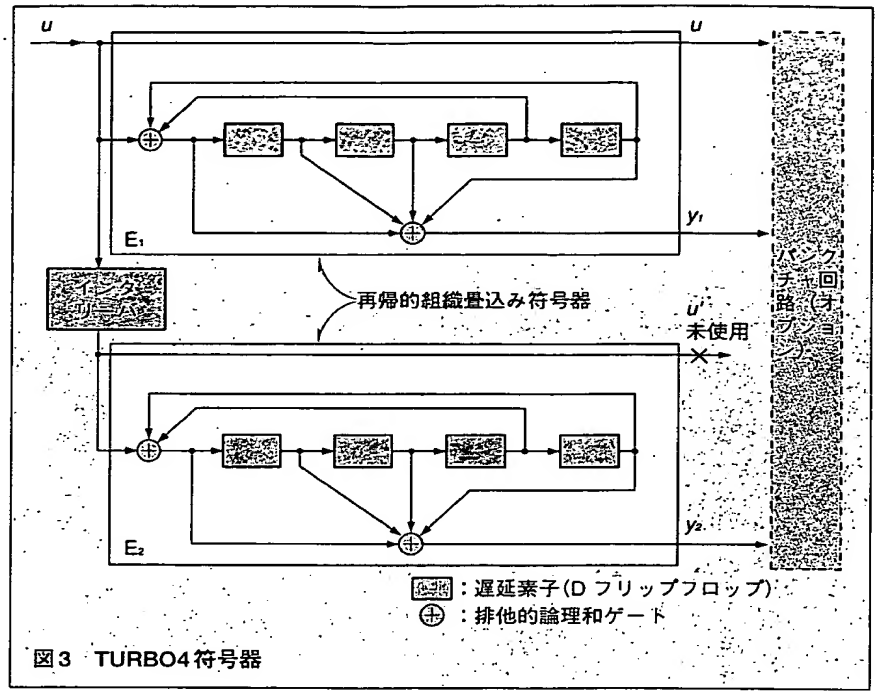


図3 TURBO4符号器

(a) 一様インタリーバ

1 A	2 F	3 K	4 P
5 B	6 G	7 L	8 Q
9 C	10 H	11 M	12 R
13 D	14 I	15 N	18 S
17 E	18 J	19 O	20 T

(b) 非一様インタリーバ

1 P	2 D	3 N	4 S
5 C	6 H	7 I	8 J
9 K	10 M	11 A	12 L
13 G	14 Q	15 E	16 B
17 T	18 R	19 O	20 F

図4 インタリーバの違い
数字順に書き込み，アルファベット順に読み出す。

畳込み符号でもブロック符号でもよい。また，ターボ符号の拡張として，要素符号を二つ以上用いる多次元ターボ符号の提案もある。

畳込み符号を要素符号とする場合，再帰的組織畳込み符号を用いることが多い。図3にTURBO4と呼ばれるターボ符号の符号化ブロック図を示す。再帰的組織畳込み符号は，表1に示す特徴がある。特に組織符号化により送信データ自身（u）も伝送することが，後述のターボ復号で説明する軟出力情報の利用に好ましいという点で，重要である。この再帰的畳込み符号は符号化率1/2，拘束長$K=5$（$\nu=4$）である。

破線のパンクチャ回路はオプションであり，図1では省略した。これがなければ，要素符号の符号器E_1，E_2からの出力y_1，y_2と，元の送信情報uの3ビットが1単位時間に送信される（したがって，符号化率1/3となる）。インタリーバからの出力u'は送信しない。y_1，y_2出力の一部分を周期的に取り除く処理をパンクチャといい，符号化率を大きくする手法としてよく使う。

畳込み符号をパンクチャしたパンクチャド（畳込み）符号の復号は，元の畳込み符号の復号に比べ，ほとんど処理は複雑にならないが，誤り訂正能力は低下する。

ターボ符号の場合，偶数奇数時刻にy_1，y_2を交代で削除すれば，符号化率は1/2になる。このようにして作られたターボ符号が図2の伝送速度0.5ビット/シンボルの例である。どの程度ビットを削除するかは実用時に選択できる。

非一様インタリーバを使う

2番目の鍵がインタリーバである[12)-16)]。インタリーバは，特定の部分に連続して発生するバースト誤りを，データの並べ替えにより訂正しやすい形の誤りとするために，誤り訂正符号と組み合わせてしばしば用いてきた。これは規則的なインタリーバだった。これに対して，ターボ符号で注目されるのはランダムな方式である。

図4を用いてターボ符号のインタリーバについて説明しよう。このインタリーバには4×5＝20個

のメモリがある（これをインタリーバ・サイズ4×5と呼ぶ）。そこに送信情報を横方向1，2，3，4…の順に書き込む。従来のインタリーバの一例である図4（a）では，縦方向A，B，C，D…の順に読み出し，要素符号の符号器E_2の入力u'に送る（これを一様インタリーバまたはブロック・インタリーバという）。

書き込みは同じだが，読み出す順を図4(b)のように変えることで，ターボ符号の特性は大きく変わる。これが非一様インタリーバ(non uniform interleaver)である。ランダム・インタリーバと呼ぶ人もいるが，実際にはある作為的な順で読み出しを定めている[注]。通信路符号化定理の証明では，ランダム符号化手法を用いている（符号長を十分長くすればランダムに選んだ符号が良い符号となる）。大きなインタリーバは，このランダム符号化手法に通じるところがあり，ターボ符号の重み分布†の改善に寄与しているといえる。

インタリーバを介して要素符号を組み合わせたターボ符号全体は，インタリーバのサイズを情報ビット数とするブロック符号と見ることができる。非一様インタリーバを適切に選べば，誤り訂正能力の指標である重み分布を改善することができる。

表2にその結果を示した。表2では，パンクチャをしない符号化率1/3のターボ符号と非組織畳込み符号を比較した。インタリーバのサイズは先の例と同じ20である。畳込み符号とターボ符号の要素符号の拘束長は3である。この条件では畳込み符号をビタビ復号する方がはるかに簡単だから，畳込み符号にとっては不利な比較である。しかし，畳込み符号，一様インタリーバを用いたターボ符号，非一様インタリーバを用いたターボ符号の順で重みの小さい符号語数が減少してゆく。

表2　重み分布

重み＼符号	畳込み符号	ターボ符号	
		一様インタリーバ	非一様インタリーバ
7	0	0	0
8	18	0	0
9	1	0	0
10	14	1	0
11	52	4	3
12	59	1	4
13	125	4	9
14	150	9	16
15	242	35	21
16	476	72	47
17	601	114	83
18	1044	200	181
19	1907	354	314
20	2551	614	611

注：畳込み符号，ターボ符号とも，符号化率1/3，拘束長3（ターボ符号は要素符号の拘束長）

ターボ復号，複数の軟出力復号器で

次に，ターボ復号の流れを図5の概念図を用いて説明しよう。U，Y_1，Y_2は誤りを含む軟判定受信データの入力である（送信データu，y_1，y_2に対応した軟判定受信データをそれぞれの大文字で表す）。以下の説明では，D_1，D_2はMAP（maximum a posteriori probability）復号器を用いるものとして説明する（MAPおよび以下の用語については，p.175の「尤度，最尤復号，軟出力復号，MAP復号」を参照）。

注）256×256の非一様インタリーバ[14]の処理を以下に説明する。

$i_t = 129\ (i+j) \bmod 256$

$j_t = [P(\xi)\cdot(j+1)] - 1 \bmod 256$

としたとき，i行，j列の順で書き込み，i_t行j_t列の順で読み出す。ただし$\xi = (i+j) \bmod 8$，$P(0)=17$，$P(1)=37$，$P(2)=19$，$P(3)=29$，$P(4)=41$，$P(5)=23$，$P(6)=13$，$P(7)=7$。(i, j, i_t, j_t=0, 1, 2, …8)。行・列は1行1列，1行2列，…1行8列，2行1列…の順。$x \bmod y$はxをyで割った剰余を表す。

†重み分布＝誤りの多い通信路では，符号のより細かな特性が影響するため，復号誤り率特性には重み分布も影響する。符号語に含まれる1の数を，この符号語の重み（RS符号などの多元符号では非零のシンボル）という。ある符号のすべての符号語に対する重みの分布を，その符号の重み分布という。小さな重みの符号語数が少ないほど，復号誤り率特性は良い。

MAP復号で軟出力を求める

アルゴリズムの詳細の説明は略すが，ターボ符号の理解のため，どのような尤度情報が扱われるかを説明しておく。

図5の復号器D_1とD_2の入出力を参照されたい。

組織符号化された情報ビット$u \in \{+1, -1\}$の対数尤度を，次式のように定義する。

$$L(u) = \log \{P(u=+1) / P(u=-1)\} \quad \cdots (1)$$

式の右辺にある$P(u=+1)$，$P(u=-1)$は，それぞれuが+1，

ターボ復号に至る，復号と符号化の歴史

情報理論の歴史は1948年シャノン（C.E.Shannon）が*Bell System Technical Journal*に“A Mathematical Theory of Communication”を発表したことに始まる（表A）。具体的な符号化方法の研究の始まりは，1950年の論文で発表されたハミング（Hamming）符号である。畳込み符号の発明は1955年である。

ディジタル・オーディオなどで広く用いられているRS（リードソロモン）符号は，構成するのが簡単であることと，効率の良い代数的な復号が行なえることが特徴である。それに対し，畳込み符号の利点は，次に述べる軟判定復号がブロック符号に比べ簡単に実現できることである。

軟判定は，誤り訂正による復号の前段の復調で行なう。復調は，受信したアナログ信号からディジタル・データの値を求める処理であり，2値の情報ならば0か1を判定する。軟判定処理をする場合，受信したデータが0（または1）である確率がどの程度であるかを**補助情報**として出力する。復号では確率そのものでなく，確率の対数をとり，定数倍し，整数で近似するなどして利用する。軟判定を行なう復調に対し，補助情報を伴わない復調は**硬判定**と呼ばれる。

最も簡単な軟判定は，0か1か曖昧（あいまい）なときは，0とも1とも判断しないで，消失（erasure）と呼ぶシンボルとする場合である。つまり，0，1および消失の3通りの判断を用いる復調だ。この方法はブロック符号の代数的復号にも利用しやすい。これを，消失訂正復号（erasure and error decoding）と言う。ただし，3通りの判断では，十分な復号特性を得られないことが多いので，消失訂正復号は**軟判定復号**に含めないこともある。軟判定復号には，通常4レベル程度の補助情報を用いる。補助情報は0，1の2通りの信号に付けられるから，この場合8値軟判定と呼ぶ。

本文中でシャノン限界の例にあげたBPSK同期検波，白色ガウス雑音通信路ならば，受信信号を分別する復調器の出力電圧値が利用できる。それを利用する復号では，符号により異なるが，硬判定の復号と比べて2～3dBの利得がある。これを**軟判定利得**と呼ぶ。この2dBが実用上大きな効果になることもある。

また，より劣悪な状況，たとえば移動体通信で，狭帯域変調を想定すれば軟判定利得は数dB～十数dB以上にもなりうる。これは，移動体通信のように受信状態が変動する場合には軟判定の補助情報が有効に利用できること，BPSKと比べ他の変調方式では誤り訂正符号化なしのビット誤り率特性の曲線が緩やかであることによる。一方，このような場合の軟判定復号には，S/Nなどの受信状態の推定が必要になるなどの問題があり，実用上はその通信方式に合った補助情報作成が鍵となる。

誤りの多い通信路では，畳込み符号と軟判定ビタビ復号の組み合わせが良いと考えられていた。さらに強力な方法として，あらかじめRS符号で符号化したデータを，さらに畳込み符号で符号化する**連接符号**が提案され，惑星探査などの深宇宙通信に利用された。受信側ではまず畳込み符号部分を軟判定ビタビ復号し，その結果からRS符号を復号する。

後段のRS符号の復号は硬判定復号であるが，これを軟判定復号にすれば性能は向上する。そのための軟判定情報を出力するのが軟

－1である確率である。

MAP復号では，情報ビットの軟出力$L(u^*)$は

$$L(u^*)=L_c \cdot U+L(u)+L_e(u) \quad \cdots (2)$$

と与えられる。ここで，$L_c \cdot U$は通信路値と呼び，軟判定復調により求まる。L_cはS/Nにより定まる定数，Uは復調器の出力値である。$L(u)$は事前情報尤度であり，情報ビットuの統計的性質から与えられる。$L_e(u)$は外部情報尤度で，検査ビットY_1，Y_2から復号により定まる量である。具体的にはビタビ復号同様に符号のトレリス線図

出力復号（soft output decoding）である。簡単に言えば，復調器が軟判定の補助情報を出力するように，誤り訂正の復号器が復号結果の信頼度に対して補助情報を出力するものだ。

J.Hagenauerらはビタビ復号法を基に軟出力復号を行なう軟出力ビタビ復号（soft-output Viterbi algorithm：SOVA）を提案した[18]。厳密な軟出力復号は計算量が大きい。したがって，簡易に適切な補助情報を得ることが軟出力復号の要点である。Hagenauerらの軟出力ビタビ復号は，従来のビタビ復号と比べさほど計算量の増加なしに実現できる。さまざまな研究者が，新たな軟出力ビタビ復号を提案している。一方ターボ符号の要素符号は簡単であるから，最適な軟出力を得るMAP復号（p.175の「尤度，最尤復号，軟出力復号，MAP復号」参照）を利用することも多い。

軟出力ビタビ復号により，連接符号後段のRS符号の復号にも軟判定情報が得られることになる。しかし，RS符号はブロック符号であり，それに適する軟判定復号として一般化最小距離復号などが提案されているものの，その性能は十分なものではない。特にRS符号は2元でなく多元符号であるため，畳込み符号の復号器の軟出力はそのまま利用できず，複数の出力の合成値という形で利用しなければならないことも問題を難しくしている。C.Berrouらは軟出力復号がもっと有効に利用できるような符号の組み合わせがあるに違いないと考え，ターボ符号・復号にたどり着いたのかもしれない。

表A，符号理論の歴史（軟判定復号を中心に）

1948	情報理論のスタート：Shannon
1950	＊ハミング符号：Hamming，単一誤り訂正ブロック符号
1955	畳込み符号：Elais
1956	＊巡回符号：Prange
1957	逐次復号：Wosencraft
1959	＊BCH符号
1960	＊RS符号
	＊Peterson復号法：BCH符号，RS符号の最初の代数的復号アルゴリズム
1963	逐次復号（Fanoアルゴリズム）
1963	しきい値復号法，APP復号法：Massey多数決を利用した簡易な硬判定・軟判定復号法
1966	連接符号：Forney
	GMD復号法：Forney，消失復号を利用したブロック符号の軟判定復号
1967	ビタビ復号：Viterbi
1968	＊BM復号法：BCH，RS符号の高速な多重誤り訂正アルゴリズム
1971	＊ゴッパ符号：Goppa
1972	＊Chaseアルゴリズム：ブロック符号の軟判定復号法
1974	MAP復号：Bahl
1975	多レベル符号化変調：今井，平川
1976	トレリス符号化変調：Ungerboeck
1978	＊ブロック符号のビタビ復号法：Wolf
1981	＊代数幾何符号：Goppa
1989	SOVA：Hagenauer
1993	ターボ符号：Berrou
1993	＊ブロック符号のトレリス簡略化：藤原ほか

＊印はブロック符号に限られた話題。

APP：a posteriori probability
BCH：Bose-Chaudhuri-Hocquengham
BM：Barlekamp-Massey
GMD：generalized minimum distance
MAP：maximum a posteriori probability
RS：Reed-Solomon
SOVA：soft-output Viterbi algorithm

を利用して計算する。詳細は略すが，MAP復号はこの$L_e(u)$の計算量が大きい。

MAP復号器[17),18)]は送信情報ビットがuであるときに，事前情報尤度$L(u)$，外部情報尤度$L_e(u)$，通信路値$L_c \cdot U$を入力し，軟出力$L(u^*)$を出力する。

復号の手順

●初回の動作

(1) 図3の再帰的組織畳込み符号器E_1に対応する軟出力復号器D_1では，Uから通信路値$L_c \cdot U$を求め入力する。さらに，データ系列Y_1も入力し，Uの軟出力$L(u^*)$を出力する。事前情報尤度の入力はない（$L(u)=0$）。$L(u^*)$はインタリーバπ_2に入力される。

(2) 次に，図3の符号器E_2に対する軟出力復号器D_2を動作させる。インタリーバπ_1から出力される系列U'は，図3の符号器E_2におけるu'と同順序である。このU'から求めた通信路値$L_c \cdot U'$を入力する。これと同期して，Y_2，$L(u)$をともに軟出力復号器D_2に入力する。事前情報尤度$L(u)$はD_1で求めた軟出力から与えられる外部情報尤度$L_e(u)=L(u^*)-L_c \cdot U$を用いる。これはインタリーバ$\pi_2$を介して与えられる（$\pi_1$の出力，つまり$U'$の順序と同じにして，$u'$を推定するため）。$\pi_1$，$\pi_2$は単なる0，1だけではなく軟出力による補助情報も含んで処理するところが，符号器のインタリーバと異なる。事前情報尤度が与えられるところを除けば動作はD_1とまったく同じであり，軟出力$L(u^*)$を出力する。1回だけで復号を終了する場合は，$L(u^*)$の正負を+1，−1と硬判定したものが復号結果となる。

●n回目の動作（$n=2, 3, \cdots$）

(3) 1回目の動作と異なり，事前情報尤度$L(u)$として，$n-1$回目のD_2の復号結果から求まる外部情報尤度$L_e(u)$を用いる。これはデインタリーバπ^{-1}により，Uと同じ並び順に変換されて復号器へ入る。デインタリーバπ^{-1}は，インタリーバπ_2で入れ替えた順番で入力されてくるデータを元に戻す，逆関数の働きをするインタリーバである。

(4) 初回の(2)と同じ操作を繰り返す。

シャノンの通信路符号化定理

通信路符号化定理はシャノンの第2定理とも言う。後述する通信路容量C（単位はビット/シンボル（bit/symbol））とすると，通信路符号化定理は以下のように述べることができる。

【通信路符号化定理】通信路容量C（ビット/シンボル）の通信路を用いて伝送速度R（ビット/シンボル）で情報を伝送するとき，誤り訂正の符号の符号長を大きくすれば，$R \leq C$ならば誤り確率をいくらでも小さくできる符号化方法が存在する。$R > C$ならば，そのような符号化法は存在しない（証明は略す）。

0，1の情報を送り，確率pでランダムに誤る最も簡単な通信路のモデルを仮定しよう。このモデルを2元対称通信路と呼ぶ。その通信路容量Cは，

$$C = 1 + \{p \log_2 p + (1-p) \log_2 (1-p)\}$$（ビット/シンボル）

で与えられる。

よく知られたBPSK（binary phase shift keying）のビット当たりの信号対雑音電力比E_b/N_0対ビット誤り率pの特性と組み合わせれば，p.166の図2に示したE_b/N_0対CのBPSK限界に近い曲線を得る。この通信路は受信側も0，1と判断する硬判定復調（p.170の「ターボ復号に至る，復号と符号化の歴史」参照）になっている。BPSK限界は，軟判定する場合に相当する2入力連続出力通信路のCの式から計算される。

通信路の違いや変調方式，復調方式の違いにより通信路容量は変わる。その最大値を結んだものがシャノン限界である。

通信路符号化定理は，具体的な符号化法は与えず，存在を示す定理である。だから，ターボ符号のようにその限界に近い性能を実現できる具体例は，大変意義深い。

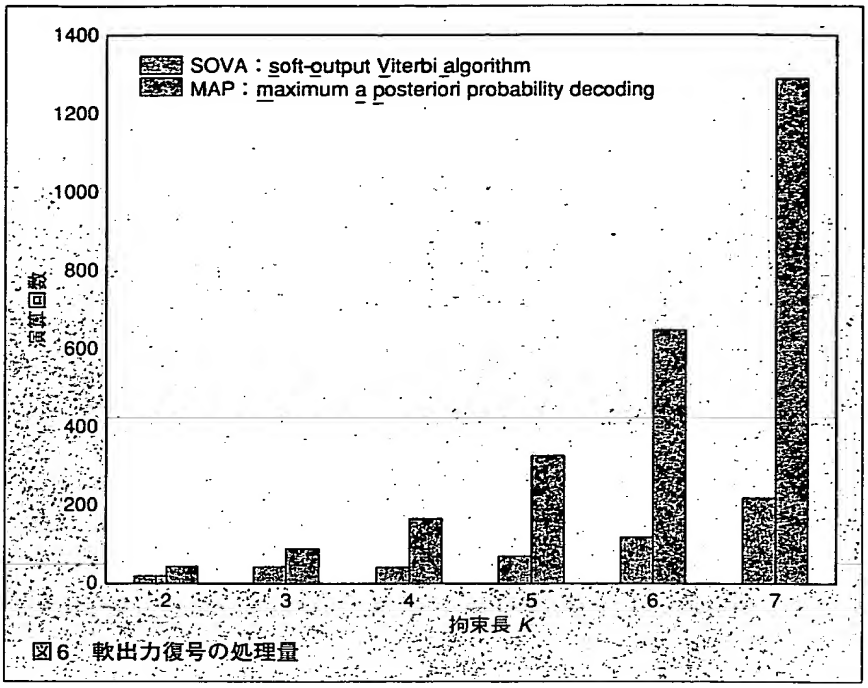


図6 軟出力復号の処理量

表3 MAPとSOVAの演算回数

演算	Log-MAP	SOVA
最大値計算	$5\times2^{(K-1)}-2$	$3K+2^K-2$
加算	$15\times2^{(K-1)}+9$	2^K+8
士符号処理	8	8
ビット比較	——	$6K$
テーブル・ルックアップ	$5\times2^{(K-1)}-2$	——

ただし，Kは畳込み符号の拘束長。演算回数は符号化率$1/n$の符号（$n=2, 3, \cdots$）の場合。

り具体的な処理項目については表3に示した。MAPとしては，尤度の対数値を用いることで乗算を加算に置き換えるLog-MAPを用いるものとした。

ターボ復号では，復号器が2個用いられ，さらに繰り返し復号するので，それも合わせて考えなければならない。拘束長が大きいと，SOVAに比べMAPは7倍近い計算量になるが，拘束長3では3倍程度，4では4倍程度となる。

図7に，SOVAを用いるターボ復号の繰り返し回数による復号誤り率特性の違いを示した。繰り返し回数を増やすことで特性が向上する。符号化なしのBPSKに比べ，10^{-5}のビット誤り率を達成するのに繰返し5回のSOVAでは7.3dB小さいE_b/N_0でよい。これを7.3dBの符号化利得があるという。MAPは5回の繰り返しのみ示すが，符号化利得は8.5dBとさらに優れている。ただし，インタリーバ・サイズや，要素符号の選択によって特性は変化する。拘束長の短い要素符号を用いて，符号化利得を上げたいならMAPを利用する。逆に拘束長の長い符号が利用可能であり，復号の繰り返し回数を増やしてもよいならば，SOVAを利用する，というように用途や状況によって答えが変わるだろう。しかし，その選択は容易ではない。移動体通信の環境では，必ずしも拘束長の長い要素符号を使うことがよい選択ではないことも報告されている[10]。

ターボ方式が向く応用先と，問題点

ここでは，ターボ符号の応用先や，LSI化にまつわる問題点をとりあげる。さらに，ますます熱気を帯びてきた研究の動向について概観してみる。

深宇宙通信，移動体通信，放送

シャノン限界に近づくことが重要な意味を持ち，ターボ符号が即有望と見込まれるのは惑星探査などの深宇宙通信である。この用途では，探査機からの受信データを一時貯え，時間をかけて復号すればよい。探査機の送信機電力，アンテナ・サイズが制限され，地球側で受信できる信号のS/Nはたいへん小さい。これまで，こうした分野はRS符号と畳込み符号の連接符号を利用してきたが，今後はターボ符号が候補となろう。その場合，強力な誤り訂正能力をもたせるために，符号化率は1/3より

さらに小さい構成が望まれる。このため，現在はRS符号と畳込み符号の連接符号を用いることが多いが，二つ以上の要素符号を用いる多次元ターボ符号の利用も有力である。

移動体通信への応用は，すでに確実とみなされる状況になった。多数の研究が進められている。ターボ符号化・復号が，フェージング通信路（受信電波の強度がめまぐるしく変動する移動体通信の通信路）に対しても，うまく誤りを訂正できるからである。

一方，(1) インタリーバ・サイズ，繰り返し復号による処理遅延の増大，(2) エラー・フロア(error floor)現象の発生，この2点をいかに解決するかが実用化への鍵である。エラー・フロア現象は，E_b/N_0が高いときの問題である。E_b/N_0が高くなるにつれビット誤り率は下るはずだ。しかし，E_b/N_0が高くなるとそれが鈍り，場合によっては，あるE_b/N_0から誤り率が変わらなくなる。そこをエラー・フロアという。原因としては，移動

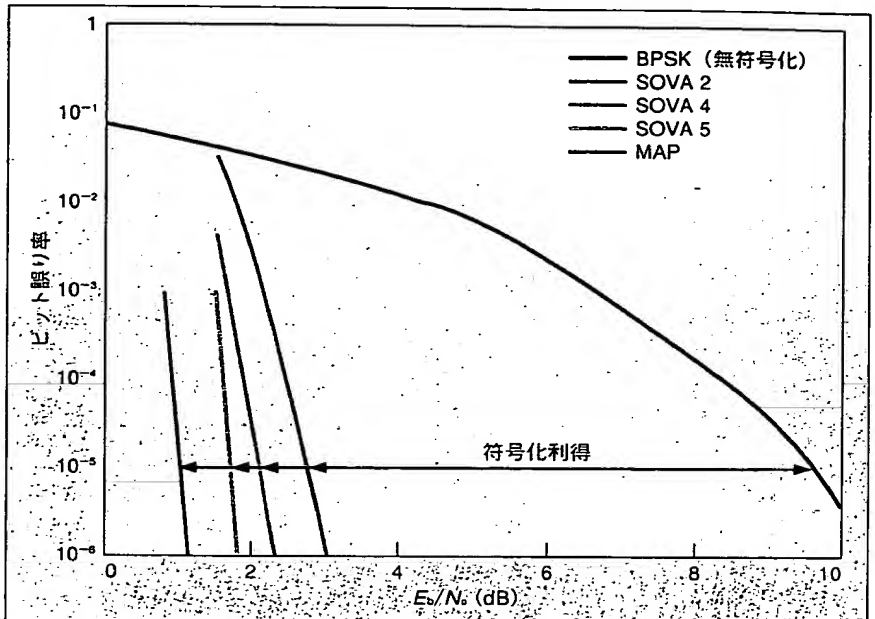


図7 繰り返し回数によるターボ復号の性能
図3のTURBO4符号器を用い，インタリーブ・サイズが64×32の場合。

尤度，最尤復号，軟出力復号，MAP復号

はじめに最尤復号とMAP復号を説明しよう。**最尤復号**（maximum likelihood decoding：ML decoding）は，受信した軟判定または硬判定情報から，最も確からしい符号語を一つ推定するという復号法である。符号語をx，受信語，すなわち誤りの加わった符号語をVとするとき，xを送信したという条件の下でVを受信する条件付き確率$p(V|x)$を最大にするx（あるいはp.164の図1の場合は情報u）を推定する復号法が最尤復号である。ここでいうVは本文で説明するターボ復号では，図1のターボ符号全体を考えれば，UおよびY_1，Y_2すべてである。D_1の復号という点から言えば，VはUおよびY_1である。これは，すべての符号語が等確率で送られるという条件の下で，正しく復号される確率を最大にするという意味で最適な復号法である。ビタビ復号はそれを具体的に実現する。

MAP復号（最大事後確率復号：maximum a posteriori probability decoding）は，正しく復号される確率を最大とする復号法である。これは，各符号語の送信確率$p(x)$が既知のときに尤度関数との積$p(x)p(V|x)$を最大とするようにxを推定する復号法と定義される。$p(x)$が等確率であるとみれば，前述の最尤復号はMAP復号に一致する。

MAP復号を実現するアルゴリズムはL. Bahlらが提案している。しかし，この方法では，uを求めるために実際にはxの中の個々のビットの尤度を計算する。このビットごとの尤度が**軟出力復号**となるのである。

体通信の通信路の性質，ターボ符号の重み分布の問題，繰り返し復号の問題などが考えられ，その改善が検討されている。

次世代移動体通信方式として世界的に標準化が進んでいるCDMA（符号分割多元接続：code division multiple access）方式は，誤りの多い環境でも高い符号化利得をもつターボ符号と組み合わせることにより，より優れた方式の実現が期待できる[10),20)~22)]。

もう一つあげられるのが放送分野である。衛星，地上波ともに今後ディジタル放送が急速に普及すると考えられるが，適切な誤り訂正方式の選択は重要な問題だ。現在，これまでの通信分野の利用と同じように，畳込み符号，RS符号，畳込み符号とRS符号の連接符号が検討されているが，ターボ符号も有力な候補となろう。処理遅延もさほど重要ではない放送分野では，十分な大きさのインタリーバと繰り返し復号により，高い符号化利得を期待できる。

ライセンスとLSI化

応用に関して重要な問題が2点ある。ライセンスとLSIである。

ターボ符号・復号に関する特許は，フランスの通信事業者であるFrance Tèlècomが取得している。実用する場合にライセンスがどのように与えられるかが気がかりな点である。

LSIの供給も，自社でターボ符号・復号器（コーデック：codec）を作成するのでなければ重要である。コーデックを選択購入できるかという点では，畳込み符号，RS符号，それらの連接符号に比べ，ターボ符号はこれからである。

現在筆者らが知るターボ符号のコーデックLSIにフランスCOMATLAS社（info@comatlas.fr）のCAS5093がある。復号データ速度は40Mビット/秒，符号化利得は7dBを実現する（符号化率1/2の場合）。0.8μmルールのCMOS技術を採用，パッケージは68ピンPLCC（plastic leaded chip carrier）。要素符号の拘束長は$K=4$，インタリーブ・サイズは32×32である。復号は4ビット軟判定のSOVAだが，実質の繰り返し2.5回をパイプラインにより実現しているので，連続的にデータを処理できる。インタリーブ・サイズや繰り返し回数など，パラメータの自由度が低いのが問題だが，今後こうしたLSIが，ほかからも出てくると考えられる。

進む理論検証/変調との一体化

応用への期待が膨らむなかで，理論的な裏付けも徐々にではあるが検証されようとしている。S. Benedettoらは二つの要素符号とインタリーバを含むターボ符号全体の構造を解析し，近似的ではあるがターボ符号の性能の裏付けを行なった[23)]。そのほかにも重み分布の解析[24)]を含め，理論的な研究が始まっている。符号化・復号の性能を高める良いインタリーバを作ろうという試み[15),16)]，要素符号の生成多項式の検討[25)]など，さまざまな研究がある。

理論，応用の両面からのとりわけ重要な課題が，ターボ符号と変調方式の一体化であるターボ符号化変調の研究である。特に伝送速度を飛躍的に向上させる多値変調との一体化が必要である[26)]。ターボ符号化変調では，要素符号や，インタリーバを変調方式に適合するように設計し直さなければならない。この分野は今後さらに発展するであろう。

誤り訂正符号の復号アルゴリズムは，信号処理のアルゴリズムとの関連が深い。相互に応用されたり，関係が明らかになったりしている。

ターボ復号も，比較的簡便なアルゴリズムで雑音を取り除けることから，磁気記録の再生系への応用[6)]，ターボ等化への利用[4),5)]，ターボ符号の復号とターボ等化を一体化するなど，提案と対象分野は広がっている。

1998年は情報理論がスタートして50周年を迎える。この8月16日から21日にかけて，ISIT '98が50周年を祝う特別テーマを含めて，Massachusetts Institute of Technologyで開催される（詳細はhttp://lids.mit.edu/isit98参照）。延べ107あるセッションのなかで，Turbo Code Performance，Applications of Turbo Codes，Iterative Decoding on Graphs，Turbo Decoding，Turbo Code Design，Iterative Decoding Performance，Interleaver Design for Turbo Codesといったセッションが予定されている。ターボ符号・復号はISIT '98における注目すべきテーマとなっている。大きなブレイクスルーを果たした発表の出現に期待したい。

参考文献

1) C. Berrou, A. Glavieux and P. Thitimajshima, "Near Shannon limit error-correcting coding and decoding : Turbo-codes," *Proc. 1993 IEEE International Conference on Communications*, pp.1064-1070.

2) T. Maseng (Ed.), "Turbo Coding Seminar," Dept. of Applied Electronics, Lund University, Aug. 1996.

3) E. Brest (Ed.), "Symposium on Turbo Codes and Related Topics," IEEE Communications Society, Aug.1997.

4) J. Hagenauer, "Source-controlled channel decoding," *IEEE Trans. on Communications*, vol.43, no.9, pp.2449-2457, Sep.1995.

5) J. Hagenauer and P. Hoeher, "A Viterbi algorithm with soft-decision outpus and its applications," *Proc. of GLOBECOM '89*, Dallas, TX, USA, pp.47.1.1-47.1.7, Nov.1989.

6) C. Heegard, "Turbo coding for magnetic recording," *Proc. for the Winter 1998 IEEE Information Theory Workshop*, San Diego, CA, USA, pp.18-19, Feb.1998.

7) 李継峰, 竹下オスカル, 今井秀樹, 「ターボ符号について」, 『電子情報通信学会技術研究報告』, IT96-45, pp.7-12, 1996年12月.

8) 山口和彦, 服部雅之, 村山淳, 「Turbo coding seminarの参加報告」, 同上, IT96-46, pp.13-20, 1996年12月.

9) R. Lumanato and H. Ogiwara, 「On the performance criterion of turbo TCM」, 『電子情報通信学会基礎・境界ソサイエティ大会論文集』, シンポジウムSA-6-1, pp.236-237, 1997年3月.

10) 藤原淳, 須田博人, 安達文幸, 「Turbo符号のW-CDMAへの適用効果」, 『電子情報通信学会技術研究報告』, SST97-77, SANE97-102, pp.19-24, 1997年12月.

11) D. Divsalar and F. Pollara, "On the design of turbo codes," JPL TDA Progress Report 42-123, May 1995.

12) E. Dunscombe and F.C. Piper, "Optimal interleaving scheme for convolutional codes," *Electron. Lett.*, vol.25, no.22, pp.1517-1518, Jun.1989.

13) P. Jung and M. I. Nasshan, "Dependance of the error performance of turbo-codes on the interleaver structure in short frame transmission systems," *Electron. Lett.*, vol.30, no.4, pp.287-288, Feb.1994.

14) C. Berrou and A. Glavieux, "Near optimum error correcting coding and decoding : turbo-codes," *IEEE Trans. Com.*, vol.44, no.10, pp.1261-1271, Oct.1996.

15) 惣道哲也, 西永望, 岩垂好裕, 「ターボコードにおけるエラーフロアを改善する交錯法に関する一検討」, 『情報理論とその応用学会, 第20回情報理論とその応用シンポジウム』, vol.20, no.1.1.4, pp.13-16, 1997年12月.

16) M. Hattori, J. Murayama and R. J. McEliece, "Pseudorandam and self-testing interleaver for turbo codes," *Proc. for the Winter 1998 IEEE Information Theory Workshop*, San Diego, CA, USA, pp.9-10, Feb.1998.

17) L. Bahl, J. Cocke, F. Jelinek and J. Raviv, "Optimal decoding of linear codes for minimizing symbol error rate," *IEEE Trans. on Information Theory*, vol.20, no.6, pp.557-567, May 1974.

18) J. Hagenauer, E. Offer and L. Papke, "Iterative decoding of binary block and convolutional codes," *IEEE Trans. Com.*, vol.42, no.2, pp.429-445, Feb.1996.

19) 山口和彦, 飯塚浩, 野村英司, 今井秀樹, 「可変しきい値APP復号法」, 『電子情報通信学会論文誌』, vol.J71-A, pp.1607-1614, 1988年8月.

20) M. C. Reed, P. D. Alexander, J. A. Asenstorfer and C. B. Schlegel, "Iterative multiuser detection for DS-CDMA with FEC," *Int. Symp. on Turbo Codes*, Brest, France, pp.162-165, Sep.1997.

21) A. Picart and R. Pyndiah, "Performance of turbo decoded product codes used in multilevel coding," *Proc. of ICC '96*, Dallas, TX, USA, pp.107-111, Jun.1996.

22) P. Jung, "Comparison of turbo-code decoders applied to short frame transmissions systems," *IEEE Journal on Selected Areas in Communications*, vol.14, no.3, pp.530-537, Apr.1996.

23) S. Benedetto and G. Montorsi, "Unveiling turbo codes : some results on parallel concatenated coding scheme," *IEEE Trans. on Information Theory*, vol.42, no.2, pp.409-428, Mar.1996.

24) Y. Svirid, "Weight distributions of turbo codes," *Proc.of 1995 IEEE International Symposium on Information Theory*, Whistler, British Columbia, Canada, p.33, Sep. 1995.

25) J. Lee and H. Imai, 「Some good polynomial for mixed-state turbo codes」, 『情報理論とその応用学会, 第20回情報理論とその応用シンポジウム』, vol.20, no.2.1.1, pp.29-33, 1997年12月.

26) S. A. Barbulescu, W. Farrell, P. Gray and M. Rice, "Bandwidth efficient turbo coding for high speed mobile satellite communications," *Int. Symp. on Turbo Codes*, Brest, France, pp.119-126, Sep. 1997.